



(11)

EP 1 313 031 A2

(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
21.05.2003 Bulletin 2003/21

(51) Int Cl.7: G06F 17/17

(21) Application number: 02256364.7

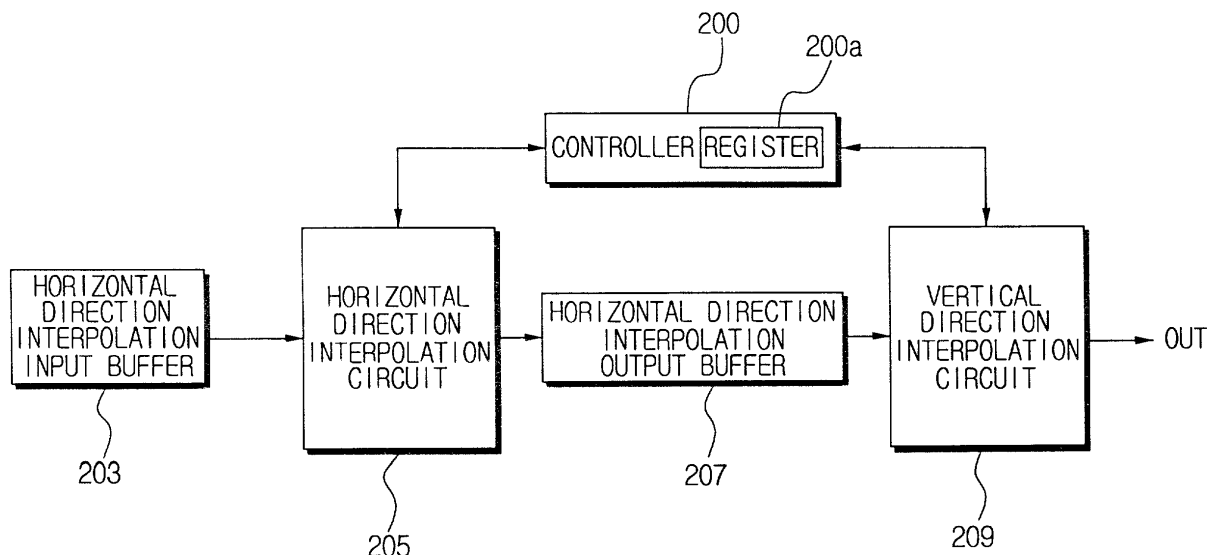
(22) Date of filing: 13.09.2002

(84) Designated Contracting States: AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LI LU MC NL PT SE SK TR Designated Extension States: AL LT LV MK RO SI	(72) Inventors: • Shim, Woo-sung Hwasung-gun, Gyunggi-do (KR) • Choi, Sung-Kyu Gwangmyung-city, Gyunggi-do (KR)
(30) Priority: 14.09.2001 KR 2001056935	(74) Representative: Geary, Stuart Lloyd et al Venner, Shipley & Co., 20 Little Britain London EC1A 7DH (GB)
(71) Applicant: SAMSUNG ELECTRONICS CO., LTD. Suwon-City, Kyungki-do (KR)	

(54) Interpolator

(57) Interpolation function coefficients used by an interpolator are calculated using functions of the interpolation interval. These function themselves have coefficients which are selected in dependence on the interpolation interval.

FIG.8



Description

[0001] The present invention relates to an interpolator for digital image processing comprising a processor for providing an interpolated pixel value from a plurality of input values according to a predetermined function and coefficient

providing means for providing a coefficient value to the processor for use in the application of said function;.

[0002] Figure 1 shows a conventional apparatus which processes digital images. For example, digital image pixel data is input from a frame store memory (not shown) to the apparatus for digital image processing. The input digital image pixel data is interpolated by the apparatus, which processes the digital image, and output to a printer (not shown) or stored in another frame store memory (not shown) for later processing.

[0003] The conventional image processing apparatus includes a controller 100, a vertical direction interpolation coefficient memory 101, a vertical direction interpolation input buffer 103, a vertical direction interpolation circuit 105, a vertical direction interpolation output buffer 107, a horizontal direction interpolation circuit 109, and a horizontal direction interpolation coefficient memory 111.

[0004] The controller 100 controls the vertical direction interpolation coefficient memory 101, the vertical direction interpolation input buffer 103, the vertical direction interpolation circuit 105, the vertical direction interpolation output buffer 107, the horizontal direction interpolation circuit 109, and the horizontal direction interpolation coefficient memory 111, so as to interpolate pixel values of the input digital image.

[0005] The vertical direction interpolation input buffer 103 temporarily stores the pixel values of the input digital image, and is embodied on a First-In First-Out (FIFO) basis. FIFO is a method of processing program operation requests which processes a first request first (i.e. a firstly-input pixel value of the digital image).

[0006] At least one interpolation coefficient, used by an equation that represents a position of an output pixel, i.e. an interpolation node, is stored in the vertical direction interpolation coefficient memory 101 and the horizontal direction interpolation coefficient memory 111. The interpolation coefficients of the respective output pixel positions are calculated according to a distance between the pixel values of the input digital image, an increment interval between the pixel positions of the output digital image, and an interpolation method in the respective output pixel positions. In addition, a series of the pixel values of the input digital image are multiplied by the corresponding interpolation coefficients, thereby interpolating the pixel values of the input digital image.

[0007] When a vertical direction interpolation processing signal is received by the controller 100, the vertical direction interpolation circuit 105 obtains vertical direction interpolation coefficients corresponding to the pixel values of the input digital image from the vertical direction interpolation coefficient memory 101, respectively multiplies the pixel values of the digital image by the vertical direction interpolation coefficients corresponding to the pixel values, and calculates a sum of the resulting values to perform the vertical direction interpolation. The pixel values of the digital image interpolated in a vertical direction by the vertical direction interpolation circuit 105 are stored in the vertical direction interpolation output buffer 107.

[0008] The horizontal direction interpolation circuit 109 receives the pixel values of the digital image interpolated in a vertical direction from the vertical direction interpolation output buffer 107. In addition, the horizontal direction interpolation circuit 109 obtains horizontal direction interpolation coefficients corresponding to the pixel values of the input digital image from the horizontal direction interpolation coefficient memory 111. Thereafter, the horizontal direction interpolation circuit 109 respectively multiplies the pixel values of the digital image interpolated in the vertical direction by the horizontal direction interpolation coefficients obtained from the horizontal direction interpolation coefficient memory 111, and calculates a sum of the resulting values to perform the horizontal direction interpolation. Accordingly, the horizontal and vertical direction interpolations of the pixel values of the input digital image are completed.

[0009] A variety of filters, such as bilinear, spline and cubic filters, which enlarge or reduce the digital image, are required to interpolate the digital image. Enlargement or reduction of the digital image by the filters should satisfy ideal filter properties. However, it is impossible to embody such an ideal filter. In Figure 1, a sinc function type filter is used, and a cubic convolution of Figure 1 uses a shape shown in Figure 2.

[0010] The convolution of Figure 2 is represented by Formula 1:

<Formula 1>

$$R_c(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1 & 0 \leq |x| \leq 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a & 1 \leq |x| \leq 2 \end{cases}$$

wherein, when 'a = -1/2' is satisfied, Formula 1 is represented by Formula 2:

<Formula 2>

$$u(x) = \begin{cases} \frac{3}{2} |x|^3 - \frac{5}{2} |x|^2 + 1 & 0 < |x| < 1 \\ -\frac{1}{2} |x|^3 + \frac{5}{2} |x|^2 - 4|x| + 2 & 1 < |x| < 2 \\ 0 & 2 < |x| \end{cases}$$

[0011] Figure 3 illustrates an interpolation method by the cubic convolution of Figure 2. Assuming that an interval of the respective periods is '1', an interpolation node Y_p is represented by Formula 3 according to a boundary condition:

<Formula 3>

$$Y_p = \begin{cases} [-\frac{1}{2} (1+t)^2 + \frac{5}{2} (1+t)^2 - 4(1+t) + 2] Y_{k-1} + \\ \quad [\frac{3}{2} t^3 - \frac{5}{2} t^2 + 1] Y_k + \\ \quad [\frac{3}{2} (1-t)^3 - \frac{5}{2} (1-t)^2 + 1] Y_{k+1} + \\ \quad [-\frac{1}{2} (2-t)^3 + \frac{5}{2} (2-t)^2 - 4(2-t) + 2] Y_{k+2} \end{cases}$$

[0012] Formula 4 is obtained by processing Formula 3 as a function of 't'. Here, Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} indicate positions of k-1, k, k+1 and k+2-th sampled output pixels of the input digital image, respectively.

<Formula 4>

$$Y_p = \{(Y_{k+2} - 3Y_{k+1} + 3Y_k - Y_{k-1})/2\}t^3 + \{(-Y_{k+2} + 4Y_{k+1} - 5Y_k + 2Y_{k-1})/2\}t^2 + \{(Y_{k+2} - 2Y_{k-1})/2\}t + Y_k$$

[0013] Formula 5 is obtained by processing Formula 4:

<Formula 5>

$$Y_p = C_1 Y_{k-1} + C_2 Y_k + C_3 Y_{k+1} + C_4 Y_{k+2}$$

$$C_1 = (-t^3 + 2t^2 - t)/2$$

$$C_2 = (3t^3 - 5t^2 + 2)/2$$

$$C_3 = (-3t^3 + 4t^2 + t)/2$$

$$C_4 = (t^3 - t^2)/2$$

[0014] Here, the respective coefficients C_1 , C_2 , C_3 and C_4 can be obtained from a lookup table.

[0015] However, when the apparatus is embodied in a form of hardware, in a scaling process such as cubic convolution, it takes a long processing time to apply the above Formula(s). In addition, as shown in Figure 4, a lot of adders

and multipliers are required to express the respective interpolation coefficients in the form of hardware. Thus, a bank is generally used as shown in Figure 5. Here, the bank is a logical unit which divides a main memory device for management. The bank is used to prepare for an increase of the main memory device or to utilize a limited memory device space by bank switching. The more banks are used, the higher the performance. Although a small number of adders and multipliers are required when the banks are used, rather than the use of the Formula(s), so far as the memory is not infinite, the bank obtains approximate values. Accordingly, such an error influences the overall image quality. Moreover, buses are used more heavily due to frequent memory accesses.

[0016] That is, in cubic convolution using N banks, the respective banks contain representative values, thus generating errors. For example, when values between 0 and 1 are divided between five banks, the values are divided at an interval of 0.2. When Y_p exists between 0 and 0.2, a distance 't' between 0 and 1 in the cubic convolution filter is set up to be 0.1, to calculate the values C_1 , C_2 , C_3 and C_4 . Accordingly, values between 0 and 0.2 are approximated to 0.1, which may generate an error.

[0017] Figures 6A, 6B, 6C and 6D illustrate an error distribution of the cubic convolution of Figure 3 and probable errors of the respective coefficients.

[0018] Referring to Figures 6A to 6D, the coefficients are divided by $1/N$ periods during N bank periods, and an error value of each period is reduced due to an increase of N. In addition, when the coefficients having errors are respectively multiplied by the pixel values, the interpolated pixel values have bigger errors. For example, assuming that 't' is 0.19 and the pixel value of the digital image represented by (255 255 1 1) belongs to a first bank, the representative value of 't' becomes 0.1, and thus the interpolated pixel value becomes 283. However, since 't' is 0.19, the interpolated pixel value calculated by using the actual 't' is 216. Accordingly, an error value of 67 is generated between the pixel value interpolated by the representative value of 't' of the bank and the pixel value interpolated by the actual 't'.

[0019] Figure 7 shows a flowchart illustrating a conventional method of processing a digital image. As shown in Figure 7 with reference to Figure 1, the controller 100 calculates a size ratio of the pixel value of the input digital image to an aimed pixel value to be interpolated by the vertical direction interpolation circuit 105 and the horizontal direction interpolation circuit 109 in operation 1001. That is, when the horizontal direction is X and the vertical direction is Y, X scale is X size of the original digital image / X size of the aimed digital image, and Y scale is Y size of the original digital image / Y size of the aimed digital image.

[0020] After calculating X and Y scales, the controller 100 measures Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} in operation 1003. Here, Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} indicate positions of the k-1, k, k+1 and k+2-th sampled output pixels of the input digital image, respectively. After measuring Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} , the controller 100 obtains 't' from the scale size in operation 1005. Here, 't' represents an X axis interval between the pixel value of the input digital image and the interpolation node, namely an interpolation interval. After calculating 't', the controller 100 obtains coefficients from the corresponding memories, the vertical direction interpolation coefficient memory 101 and the horizontal direction interpolation coefficient memory 111 according to 't' in operation 1007. As described above, the vertical direction interpolation coefficient memory 101 and the horizontal direction interpolation coefficient memory 111 store the coefficients for variables which indicate the positions of the output pixels sampled in the interpolation node calculation equation.

[0021] The controller 100 calculates the interpolation node Y_p according to the obtained coefficient in operation 1009. In addition, the controller 100 interpolates the input digital image by using the calculated interpolation node in operation 1011. Accordingly, interpolation of the input digital image is completed.

[0022] However, the conventional apparatus which processes the digital image and the method therefor require many adders and multipliers in a form of hardware, and also require many banks to obtain precise values. Moreover, when the number of banks is restricted, the respective banks store imprecise representative values for the interpolation intervals. As a result, errors are generated between the theoretical interpolated pixel values and those actually obtained and used.

[0023] An interpolator according to the present invention is characterised in that the coefficient providing means is configured to calculate said coefficient as a coefficient calculation function of the interpolation interval, said coefficient calculation function itself being selected in dependence on the interpolation interval.

[0024] Preferably, said coefficient calculation function is of the form:

$$C_n(t) = a_{kn}t + b_{kn}$$

where C_n is the coefficient, t is the interpolation interval and a_{kn} and b_{kn} are themselves coefficients, k indicating the sub-division, of the period between the pixels between which interpolation is being performed, which contains t.

[0025] Preferably, said predetermined function is a cubic function.

[0026] Other optional and preferred features are set forth in claims 4 to 24 appended hereto.

[0027] An embodiment of the present invention will now be described, by way of example, with reference to Figures 8 to 10 of the accompanying drawings, in which:

Figure 1 is a schematic block diagram illustrating a conventional digital image processing apparatus;
 Figure 2 is a graph illustrating waveforms of a cubic convolution of the apparatus shown in Figure 1;
 Figure 3 is a graph explaining an interpolation method by the cubic convolution of Figure 2;
 Figure 4 is a diagram illustrating an example where one of the coefficients for an equation representing a position
 of an output pixel in the graph of Figure 3 is embodied using adders and multipliers;
 Figure 5 is a diagram illustrating an example where the coefficient of the cubic convolution of Figure 3 is embodied
 using banks;
 Figures 6A, 6B, 6C and 6D are diagrams illustrating error distributions of the cubic convolution of Figure 3;
 Figure 7 is a flowchart illustrating a conventional method of processing a digital image;
 Figure 8 is a schematic block diagram illustrating a digital image processing apparatus according to the present
 invention;
 Figures 9A, 9B, 9C and 9D are diagrams illustrating linear approximations to the errors where a coefficient of a
 cubic convolution performed the apparatus shown in Figure 7 provided from banks; and
 Figure 10 is a flowchart illustrating a method of processing a digital image according to the present invention.

[0028] Referring to Figure 8, a digital image processing apparatus includes a controller 200, a horizontal direction interpolation input buffer 203, a horizontal direction interpolation circuit 205, a horizontal direction interpolation output buffer 207, and a vertical direction interpolation circuit 209. The controller 200 includes a register 200a.

[0029] The controller 200 controls the horizontal direction interpolation input buffer 203, the horizontal direction interpolation circuit 205, the horizontal direction interpolation output buffer 207, and the vertical direction interpolation circuit 209, so as to interpolate between pixel values of the input digital image. In addition, the controller 200 measures an interpolation interval (t) for the pixel values of the input digital image, and performs operations for interpolating the digital image by using the measured interpolation interval.

[0030] Since the horizontal direction interpolation input buffer 203, the horizontal direction interpolation circuit 205, the horizontal direction interpolation output buffer 207, and the vertical direction interpolation circuit 209 have similar functions as those shown in Figure 1, explanations thereof are omitted to avoid repetition. However, the respective elements 203, 205, 207 and 209 have different processing procedures.

[0031] The register 200a is installed in the controller 200, and stores a coefficient equation. As illustrated in Figures 9A, 9B, 9C and 9D, the coefficient equation uses a relationship that errors have linearity when coefficients of a cubic convolution are embodied as banks. Here, Figures 9A through 9D are graphs showing C_1 , C_2 , C_3 and C_4 , respectively.

[0032] That is, when a period for an X scale is divided by N banks, the period is divided into N periods. A range of the errors in the respective N periods is linear. When N is increased, the range of the errors has high linearity. As shown in Figures 9A through 9D, the errors of each period are represented by linear equations. A slight difference may be generated from the actual value due to linear expression as in the second period of C_1 and the fourth period of C_4 . However, values of C_1 and C_4 are smaller than those of C_2 and C_3 , and thus can be ignored. Moreover, the values of C_1 or C_4 have high linearity due to an increase of N.

[0033] When the number of banks is N, as many linear equations as $N \times$ sampling number for the pixel values of the input digital image may be required. When the bank for the interpolation is determined, coefficients approximating cubic convolution functions in the respective banks, can be obtained by using the linear equations. That is, the linear equation is represented by 'aX + b'. Here, 'a' denotes a gradient and 'b' denotes an Y intercept. X is decided on the basis of the scale size. Thus, X has a value less than 1 in interpolation, and greater than 1 in decimation. Since the bank can be informed by using the interpolation interval (t) decided by the scale size, the error value can be reduced to a minimum value by using the previously-decided linear equation of the bank.

[0034] The above coefficient equation satisfies and is represented by Formula 7:

<Formula 7>

$$C_n(t) = a_{kn}t + b_{kn}$$

where C_n denotes the coefficient of a variable of a position of an output pixel indicating an nth input digital image in an equation representing the position of the output pixel which is the interpolation node, t denotes the interpolation interval, a_{kn} is stored in a kth register and represents a gradient of a straight line between the position of an output pixel of an n-1th input digital image and the position of the output pixel of the nth input digital image with respect to a predetermined interpolation interval, b_{kn} is stored in the kth register and represents an intersecting point between the straight line between the position of the output pixel of the n-1th input digital image and the position of the output pixel of the nth input digital image with respect to the predetermined interpolation interval and Y axis, and 'k = 1 2, 3, ..., k, ...' and 'n = 2, 3, 4, ..., n, ...'

[0035] Referring to Figure 10, the controller 200 calculates a size ratio of the pixel value of the input digital image to an aimed pixel value to be interpolated by the horizontal direction interpolation circuit 205 and the vertical direction interpolation circuit 209 in operation 2001. That is, when the horizontal direction is X and the vertical direction is Y, X scaling factor is X size of the original digital image / X size of the aimed digital image, and Y scaling factor is Y size of the original digital image / Y size of the aimed digital image.

[0036] After calculating the X and Y scaling factors, the controller 200 measures Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} in operation 2003. Here, Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} indicate positions of the k-1, k, k+1 and k+2th sampled pixels of the input digital image, respectively. After measuring Y_{k-1} , Y_k , Y_{k+1} and Y_{k+2} , the controller 200 obtains 't' from the scale size in operation 2005. Here, 't' represents an X axis interval between the pixel value of the input digital image and the interpolation node, namely an interpolation interval. After calculating 't', the controller 200 obtains a corresponding coefficient equation from the register 200a according to 't' in operation 2007. As described above, the register 200a stores the coefficient equations for each coefficient in the interpolation node calculation equation, and is provided in the controller 200. The amount of information stored in the register 200a corresponds to the predetermined coefficient equations, thereby reducing a processing time of the controller 200.

[0037] The controller 200 calculates the coefficient from the coefficient equation in operation 2009, and obtains the interpolation node Y_p by using the obtained coefficient in operation 2011. That is, the controller 200 transmits an interpolation node calculation signal to the horizontal direction interpolation circuit 205 and the vertical direction interpolation circuit 209, multiplies the coefficients from the register 200a by the pixel values of the input digital image, respectively, and calculates a sum of the resulting values, thereby performing the horizontal and vertical direction interpolations in operation 2013. The interpolation node calculation equation satisfies and is represented by Formula 8:

<Formula 8>

$$Y_p = Y_{n-1} C_{n-1} + Y_n C_n + Y_{n+1} C_{n+1} + Y_{n+2} C_{n+2}$$

where Y_p denotes the interpolation node of the input digital image, Y_n denotes a position of an n-th sampled output pixel of the input digital image, C_n is the coefficient of the position of the nth sampled pixel of the input digital image, and 'n = 2, 3, 4, ..., n, ...'

[0038] The controller 200 interpolates the input digital image by using the calculated interpolation node in operation 2130. Accordingly, interpolation of the pixel values of the input digital image is completed.

[0039] The coefficient equation is represented by a linear equation according to the relationship that the range of the errors by N banks has linearity, thereby reducing the error of the interpolated value of the pixel value of the input digital image. In addition, when the present apparatus is embodied in a form of hardware, a large number of adders and multipliers is not needed, further simplifying the structure of the digital image processing apparatus. In the method of calculating the coefficient from the register 200a, storing the coefficient equation and obtaining the interpolation node, the coefficient equation is stored in the register 200a of the controller 200 to reduce the processing time of the controller 200.

Claims

1. An interpolator for digital image processing comprising:

a processor (205, 209) for providing an interpolated pixel value from a plurality of input values according to a predetermined function; and
coefficient providing means (200) for providing a coefficient value to the processor for use in the application of said function;

characterised in that

the coefficient providing means (200) is configured to calculate said coefficient as a coefficient calculation function of the interpolation interval, said coefficient calculation function itself being selected in dependence on the interpolation interval.

2. An interpolator according to claim 1, wherein said coefficient calculation function is of the form:

$$C_n(t) = a_{kn}t + b_{kn}$$

where C_n is the coefficient, t is the interpolation interval and a_{kn} and b_{kn} are themselves coefficients, k indicating the sub-division, of the period between the pixels between which interpolation is being performed, which contains t .

3. An interpolator according to claim 1 or 2, wherein said predetermined function is a cubic function.

4. An apparatus for processing a digital image, comprising:

a register which stores a coefficient equation;
an interpolation processing unit which interpolates the input digital image; and
a controller which:

measures an interpolation interval of pixel values of the digital image,
calculates a coefficient by substituting the interpolation interval for the coefficient equation stored in the register,
calculates an interpolation node for the digital image by substituting the coefficient, and an output pixel position value of the digital image for an interpolation node calculation equation, and
controls the interpolation processing unit to interpolate the digital image to the interpolation node.

5. The apparatus of claim 4, wherein the coefficient equation satisfies the following relationship:

$$C_n(t) = a_{kn}t + b_{kn}$$

where C_n denotes the coefficient of a variable of a position of an output pixel indicating an n -th input digital image in an equation representing the position of the output pixel which is the interpolation node, t denotes the interpolation interval, a_{kn} is stored in a k -th register and represents a gradient of a straight line passing a position of an output pixel of an $n-1$ th input digital image and the position of the output pixel of the n -th input digital image with respect to a predetermined interpolation interval, b_{kn} is stored in the k -th register and represents an intersecting point between the straight line passing the position of the output pixel of the $n-1$ th input digital image and the position of the output pixel of the n -th input digital image with respect to the predetermined interpolation interval and Y axis, and ' $k = 1, 2, 3, \dots, k, \dots$ ' and ' $n = 2, 3, 4, \dots, n, \dots$ '

6. The apparatus of claim 5, wherein the interpolation node calculation equation satisfies the following relationship:

$$Y_p = Y_{n-1} C_{n-1} + Y_n C_n + Y_{n+1} C_{n+1} + Y_{n+2} C_{n+2}$$

where Y_p denotes the interpolation node of the input digital image, Y_n denotes a position of an n -th sampled output pixel of the input digital image, and C_n is the coefficient of the position of the n -th sampled output pixel of the input digital image.

7. A method of processing a digital image, the method comprising:

measuring an interpolation interval of pixel values of the input digital image; calculating a coefficient by substituting the interpolation interval for a coefficient equation stored in a register;
calculating an interpolation node for the digital image by substituting the coefficient and an output pixel position value of the digital image for an interpolation node calculation equation; and
controlling an interpolation processing unit to interpolate the digital image to the interpolation node.

8. The method of claim 7, wherein the coefficient equation satisfies the following relationship:

$$C_n(t) = a_{kn}t + b_{kn}$$

where C_n denotes the coefficient of a variable of a position of an output pixel indicating an n -th input digital image in an equation representing the position of the output pixel which is the interpolation node, t denotes the interpolation interval, a_{kn} is stored in a k -th register and represents a gradient of a straight line passing a position of an output pixel of an $n-1$ th input digital image and the position of the output pixel of the n -th input digital image with respect

to a predetermined interpolation interval, b_{kn} is stored in the k-th register and represents an intersecting point between the straight line passing the position of the output pixel of the n-1th input digital image and the position of the output pixel of the n-th input digital image with respect to the predetermined interpolation interval and Y axis, and 'k = 1, 2, 3, ..., k, ...' and 'n = 2, 3, 4, ..., n, ...'

9. The method of claim 8, wherein the interpolation node calculation equation satisfies the following relationship:

$$Y_p = Y_{n-1} C_{n-1} + Y_n C_n + Y_{n+1} C_{n+1} + Y_{n+2} C_{n+2}$$

where Y_p denotes the interpolation node of the input digital image, Y_n denotes a position of an n-th sampled output pixel of the input digital image, and C_n is the coefficient of the position of the n-th sampled output pixel of the input digital image.

10. The method of claim 9, wherein the coefficient equation is stored in the register and has linearity so as to reduce a processing time of the digital image and prevent errors in the interpolated digital image.

11. The method of claim 7, wherein the measuring of the interpolation interval comprises:

calculating a size ratio of the pixel values of the input digital image to aimed pixel values;
measuring positions of sampled output pixels of the input digital image; and obtaining the interpolation interval according to the size ratio and the positions of the sampled output pixels.

12. The method of claim 11, further comprising selecting the coefficient equation according to the interpolation interval among stored coefficient equations in the register for each coefficient in the interpolation node calculation equation.

13. The method of claim 12, wherein the coefficient equation is stored in the register and has linearity so as to reduce a processing time of the digital image and prevent errors in the interpolated digital image.

14. The apparatus of claim 1, further comprising:

a horizontal direction interpolation input buffer which stores the pixel values of the input digital image;
a horizontal direction interpolation circuit which performs a horizontal direction interpolation to the input digital image;
a horizontal direction interpolation output buffer which stores the pixel values of the input digital image that are interpolated in the horizontal direction interpolation circuit; and
vertical direction interpolation circuit which performs a vertical direction interpolation to the input digital image.

15. The apparatus of claim 14, wherein the controller selects the coefficient equation according to the interpolation interval among stored coefficient equations in the register for each coefficient in the interpolation node calculation equation.

16. The apparatus of claim 15, wherein the coefficient equation is stored in the register and has linearity so as to reduce a processing time of the digital image and prevent errors in the interpolated digital image.

17. The apparatus of claim 4, wherein the coefficient equation and errors of the coefficient equation have linearity so as to reduce a number of adders and multipliers, where the coefficient equation is embodied in a form of hardware.

18. The apparatus of claim 4, wherein the register is stored in the controller.

19. The apparatus of claim 7, wherein the coefficient equation is stored in the register and has linearity so as to reduce a processing time of the digital image and prevent errors in the interpolated digital image.

20. A computer readable medium encoded with operating instructions for implementing a method of processing a digital image by a computer, the method comprising:

measuring an interpolation interval of pixel values of the input digital image; calculating a coefficient by substituting the interpolation interval for a coefficient equation stored in a register data set;

calculating an interpolation node for the digital image by substituting the coefficient and an output pixel position value of the digital image for an interpolation node calculation equation; and controlling an interpolation processing unit to interpolate the digital image to the interpolation node.

21. The computer readable medium of claim 20, wherein the measuring of the interpolation interval comprises:

calculating a size ratio of the pixel values of the input digital image to aimed pixel values;
measuring positions of sampled output pixels of the input digital image; and obtaining the interpolation interval according to the size ratio and the positions of the sampled output pixels.

22. The computer readable medium of claim 20, further comprising selecting the coefficient equation according to the interpolation interval among stored coefficient equations in the register data set for each coefficient in the interpolation node calculation equation.

23. The computer readable medium of claim 20, wherein the coefficient equation satisfies the following relationship:

$$C_n(t)=a_{kn}t+b_{kn}$$

where C_n denotes the coefficient of a variable of a position of an output pixel indicating an n-th input digital image in an equation representing the position of the output pixel which is the interpolation node, t denotes the interpolation interval, a_{kn} is stored in a k-th register and represents a gradient of a straight line passing a position of an output pixel of an n-1th input digital image and the position of the output pixel of the n-th input digital image with respect to a predetermined interpolation interval, b_{kn} is stored in the k-th register and represents an intersecting point between the straight line passing the position of the output pixel of the n-1th input digital image and the position of the output pixel of the n-th input digital image with respect to the predetermined interpolation interval and Y axis, and 'k = 1, 2, 3, ..., k, ...' and 'n = 2, 3, 4, ..., n,'

24. The computer readable medium of claim 23, wherein the interpolation node calculation equation satisfies the following relationship:

$$Y_p=Y_{n-1}C_{n-1}+Y_nC_n+Y_{n+1}C_{n+1}+Y_{n+2}C_{n+2}$$

where Y_p denotes the interpolation node of the input digital image, Y_n denotes a position of an n-th sampled output pixel of the input digital image, and C_n is the coefficient of the position of the n-th sampled output pixel of the input digital image.

FIG. 1

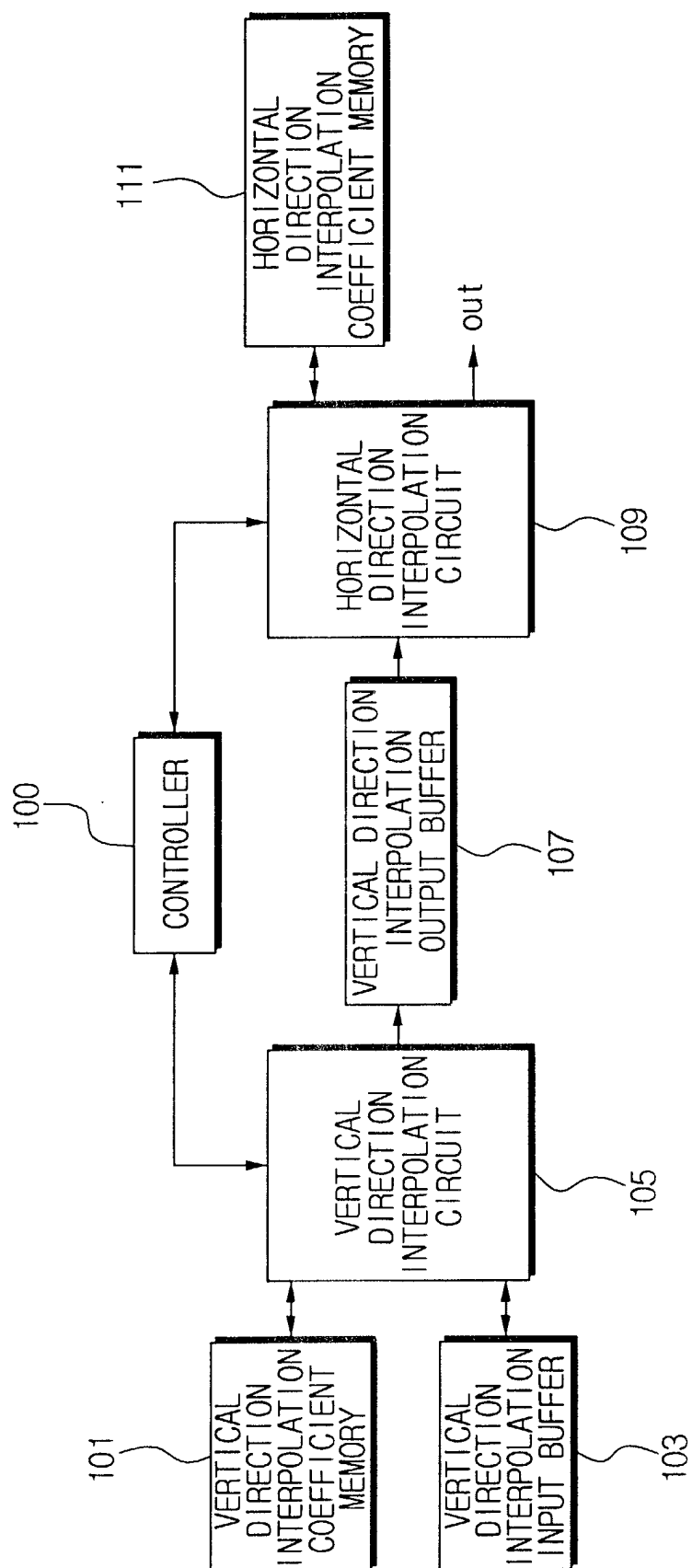


FIG.2

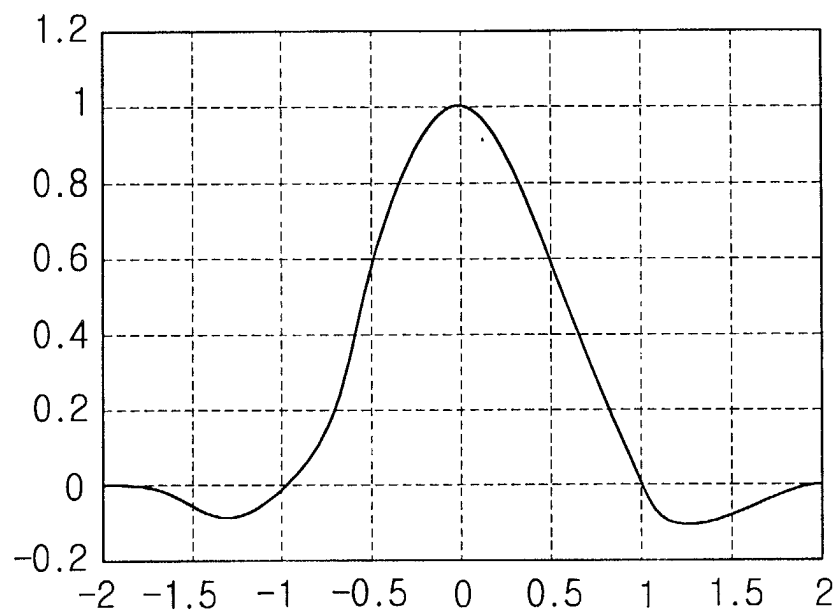


FIG.3

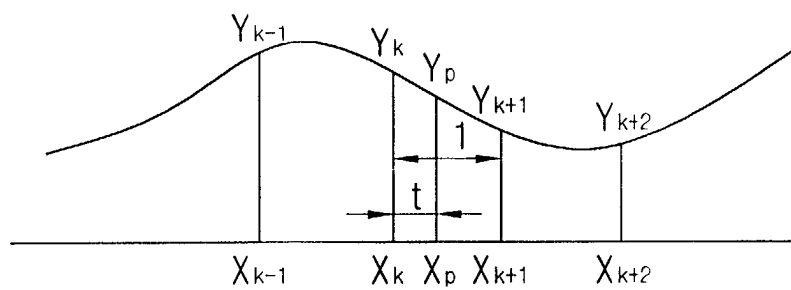


FIG.4

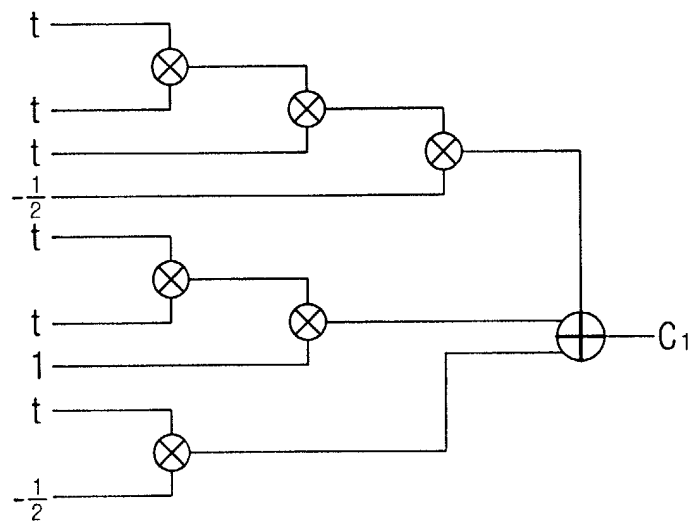


FIG.5

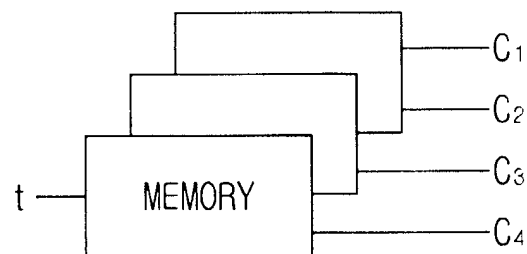


FIG.6A

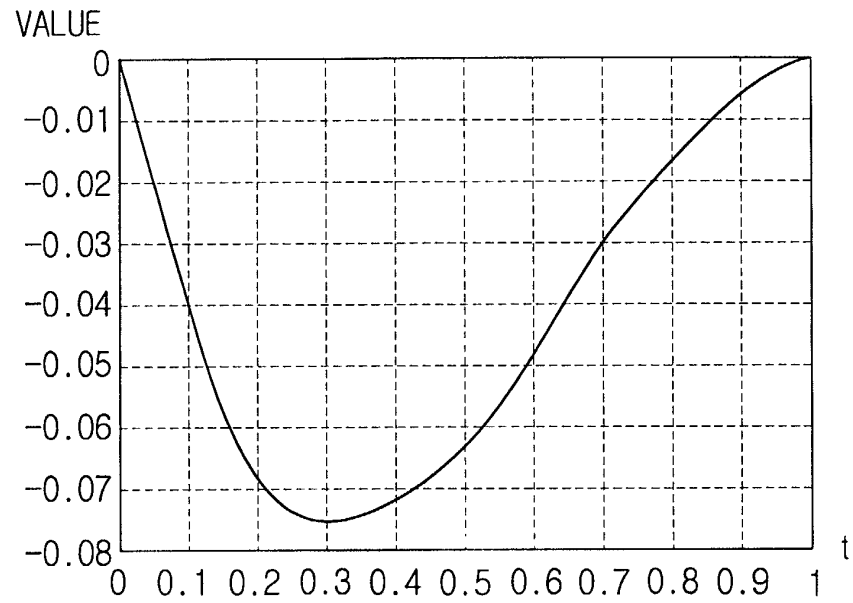


FIG.6B

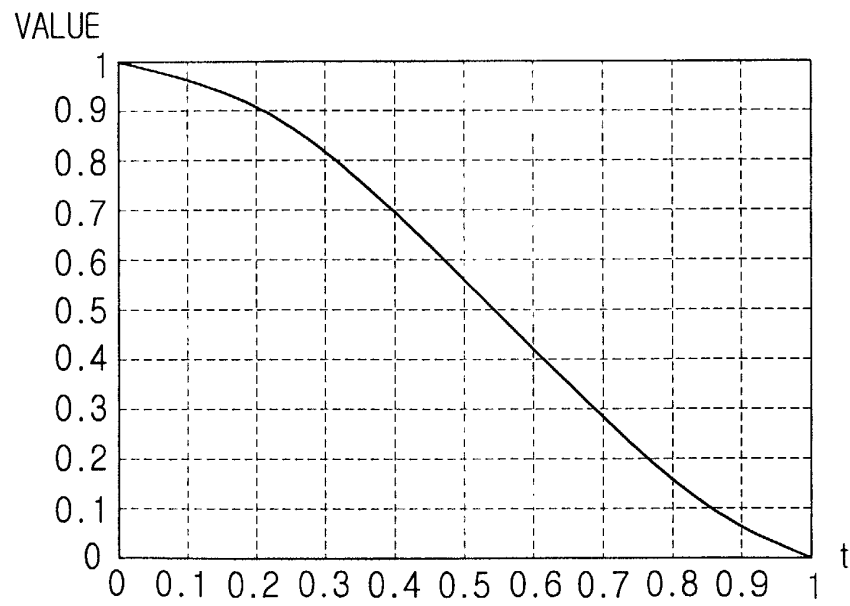


FIG.6C

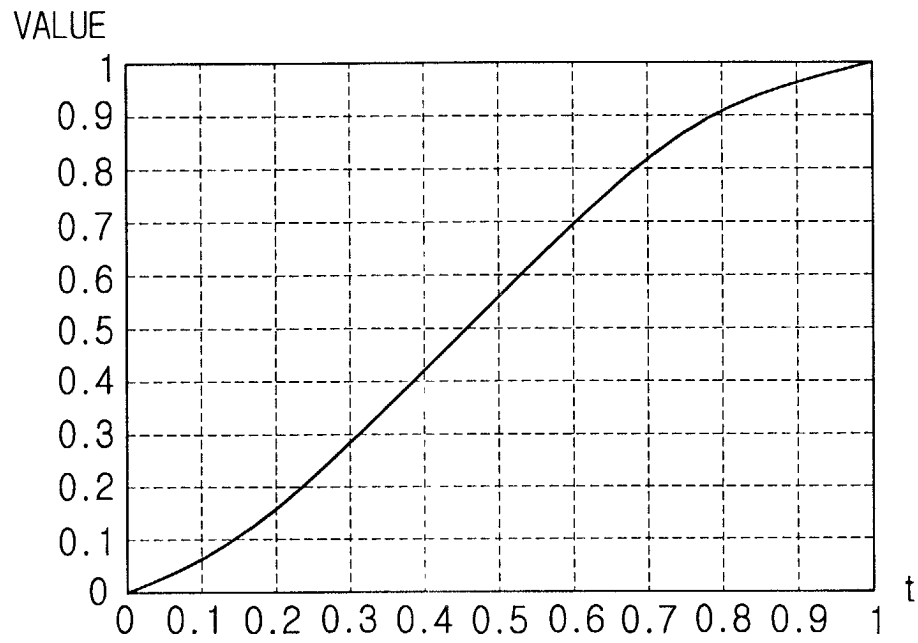


FIG.6D

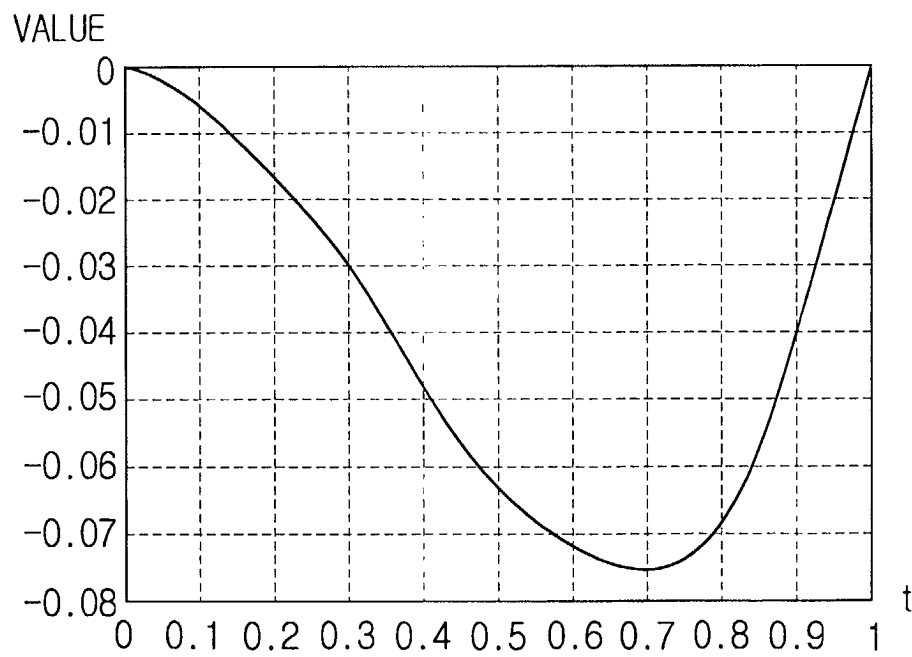


FIG. 7

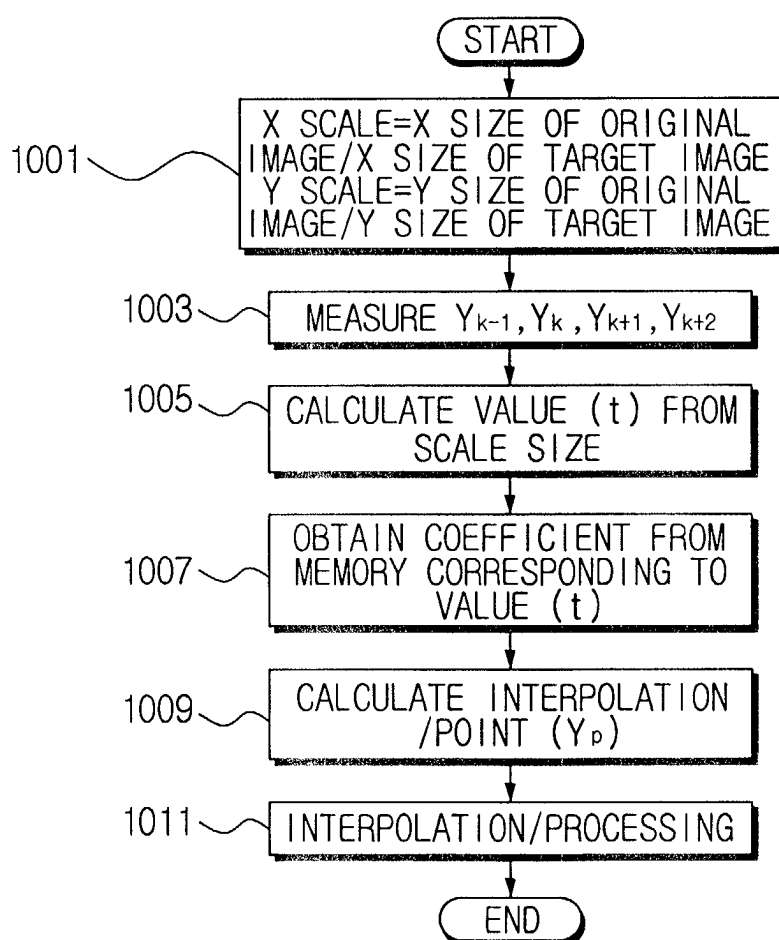


FIG. 8

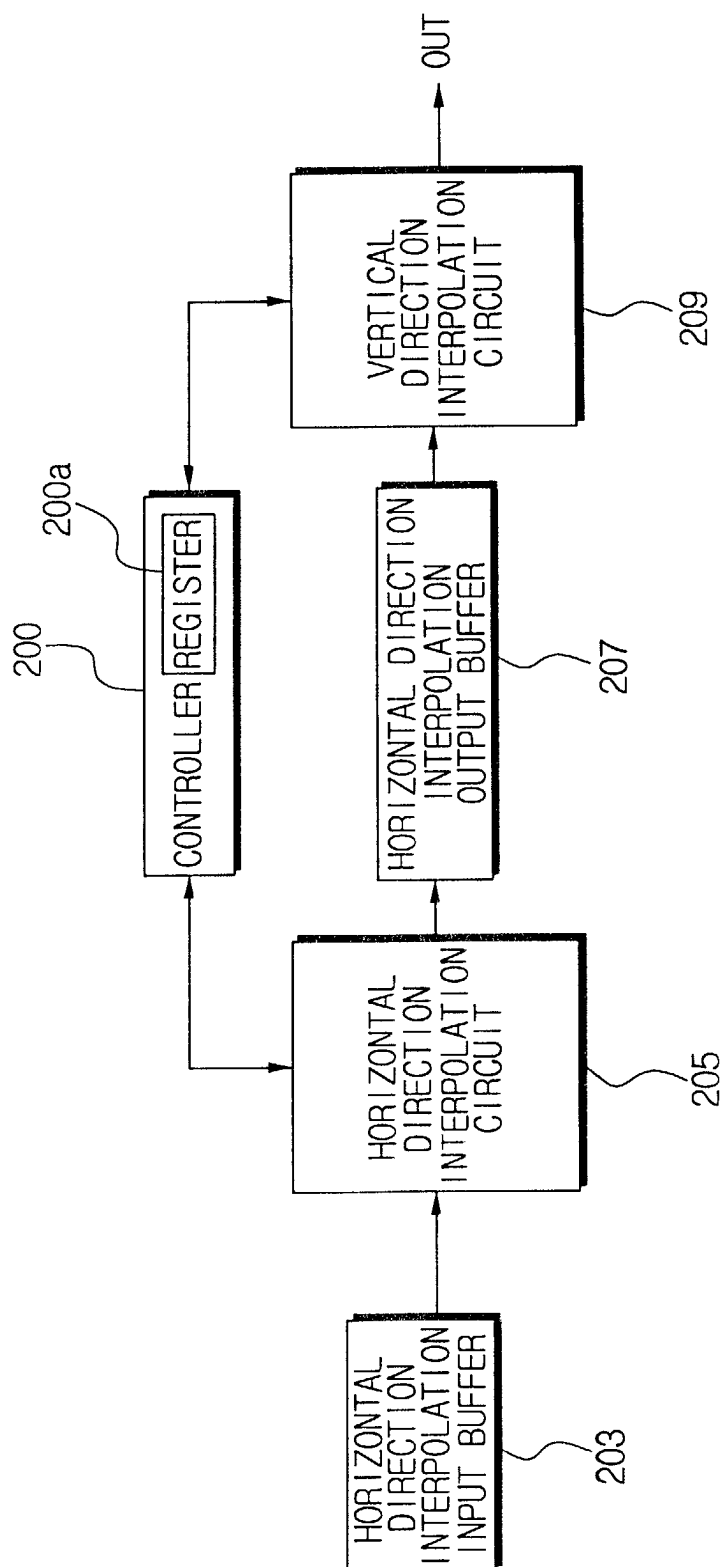


FIG.9A

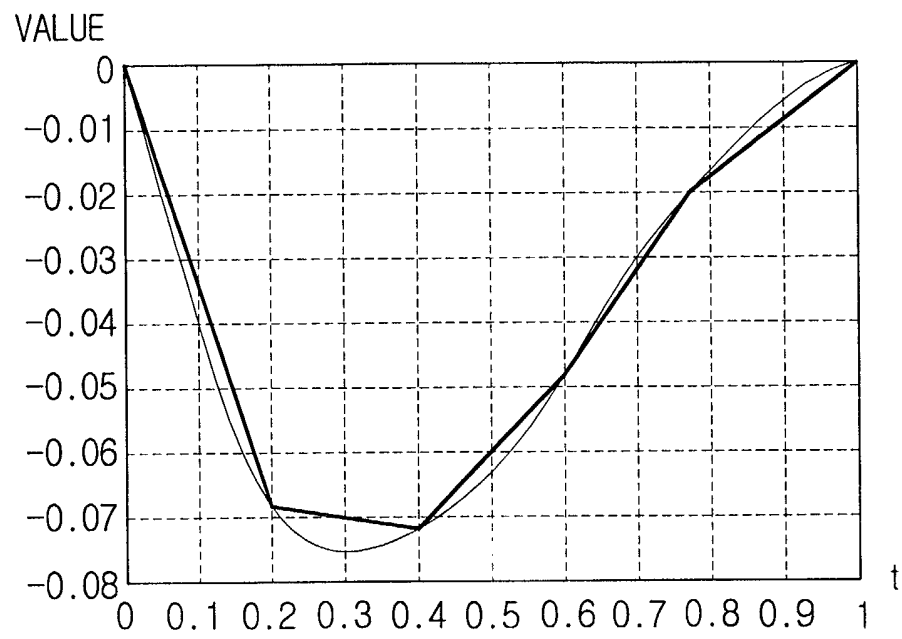


FIG.9B

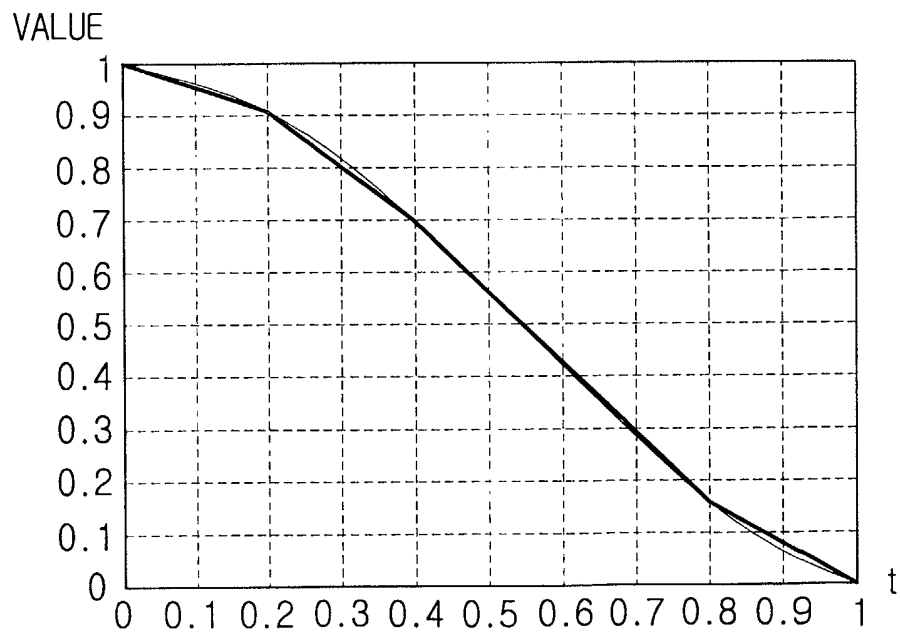


FIG.9C

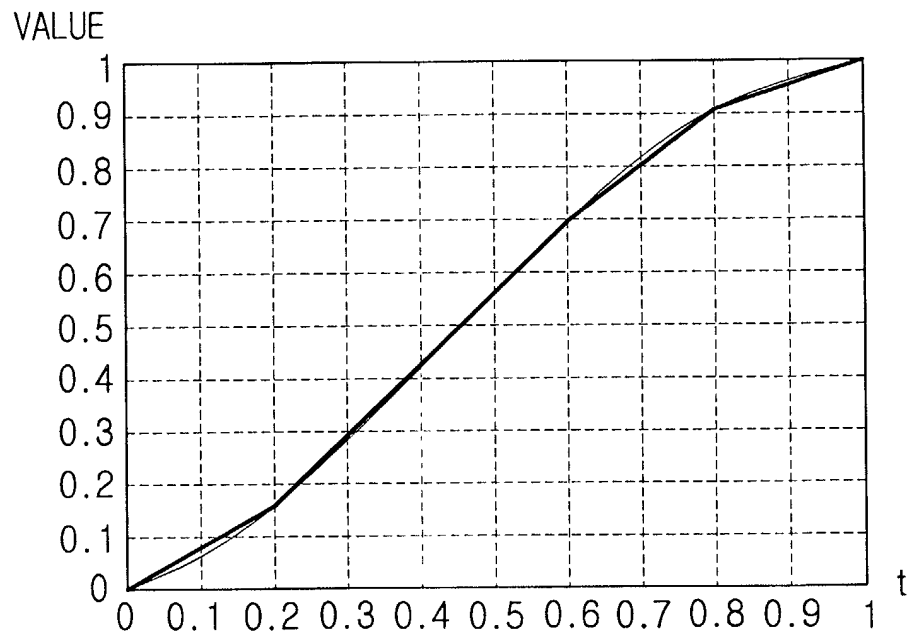


FIG.9D

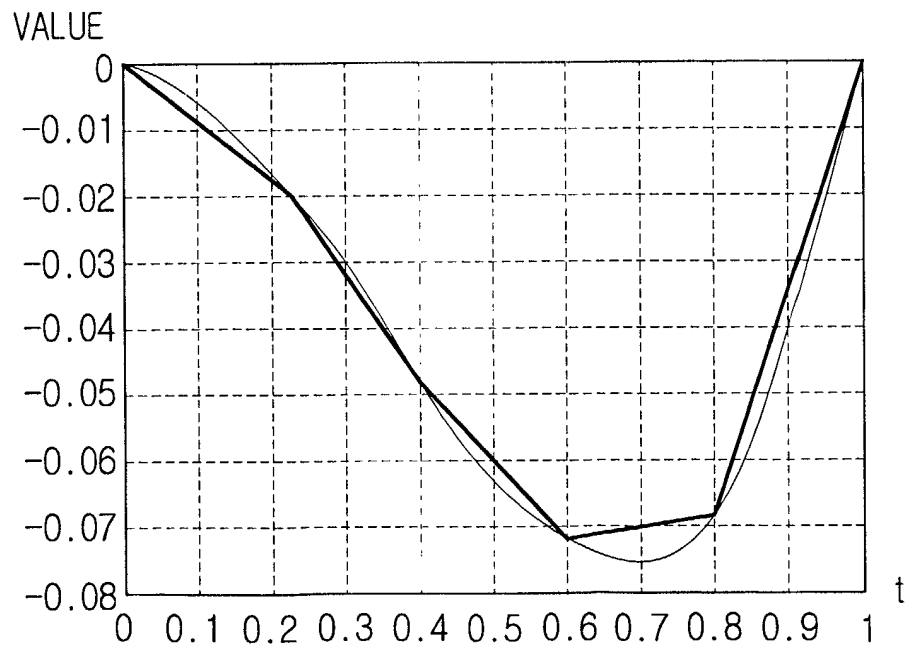


FIG.10

